

共 43 题 当前题型：问答题(问答题)

设一幅图像由物体和背景构成，设背景的平均灰度和方差为 $\mu_1 = 30$ ， $\sigma_1^2 = 900$ ，物体的平均灰度和方差为 $\mu_2 = 90$ ， $\sigma_2^2 = 400$ ，对于如下的两种情况，绘出灰度直方图并确定出将物体从背景中分割出来的最优阈值。假设物体占据了图像的如下百分比：

(a) 30%

(b) 60%

41.

(本题分数:10 分) ☐ 存疑

提交附件：

浏览 上传

上一题

下一题

$$E(aX+bY)=aE(X)+bE(Y);$$

$$D(aX+bY)=a^2D(X)+2abCov(X,Y)+b^2D(Y);$$

解答：

$$(1) 1. E(0.7X+0.3Y)=0.7E(X)+0.3E(Y)=0.7*30+0.3*90=48$$

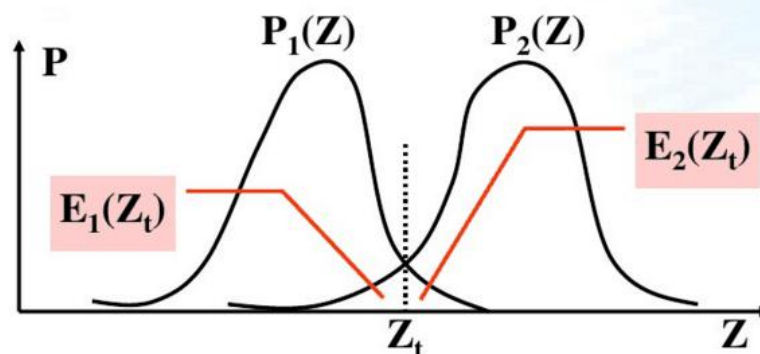
$$D(0.7X+0.3Y) = 0.7^2D(X)+0.3^2D(Y)=0.49*900+0.09*400=477$$

2. 图（3sigma 原则）

直方图图像类似下面：

如果设 $\theta=1-\theta$ ，即 $\theta=1/2$ 时，

$$Z_t = \frac{\mu_1 + \mu_2}{2}$$



从前面可以看出，假如图像的目标物和背景像素灰度级概率呈正态分布，且偏差相等($\sigma_1^2 = \sigma_2^2$)，背景和目標物像素总数也相等($\theta = 1/2$)，则这个图像的最佳分割阈值就是目标物和背景像素灰度级两个均值的平均

3. 最优阈值(48)

$$(2) E(0.4X+0.6Y)=0.4E(X)+0.6E(Y)=0.4*30+0.6*90=66$$

$$D(0.4X+0.6Y) = 0.4^2D(X)+0.6^2D(Y)=0.16*900+0.36*400=288$$

最优阈值（66）

直方图图像类似

关于最佳阈值（大津法）的求解：

将上两式代入，且对两边求对数，得到：

$$\ln\sigma_1 + \ln(1-\theta) - \frac{(Z_t - \mu_2)^2}{2\sigma_2^2} = \ln\sigma_2 + \ln\theta - \frac{(Z_t - \mu_1)^2}{2\sigma_1^2}$$

简化为： $AZ_t^2 + BZ_t + C = 0$

上式是 Z_t 的一个二次方程式，有两个解，因此，要使分割误差最小，需要设置两个阈值，即上式的两个解。如果设 $\sigma^2 = \sigma_1^2 = \sigma_2^2$ ，即方差相等，则上式方程存在唯一解，即：

$$Z_t = \frac{\mu_1 + \mu_2}{2} + \frac{\sigma^2}{\mu_1 - \mu_2} \ln\left(\frac{1-\theta}{\theta}\right)$$


可以使用这张 PPT（全部的 PPT 找不到了）上面的公式求解，但是对数运算太麻烦，人工计算代码麻烦，所以编程实现求该值，程序如下：

```
// CBestThreshold.cpp : 定义控制台应用程序的入口点。
//
#include "stdafx.h"
#include "math.h"
int _tmain(int argc, _TCHAR* argv[])
{
    double Threshold;
    for(Threshold=0.0;Threshold<255;Threshold=Threshold+0.1)
    {


        //if(abs((1.946+(Threshold-30)*(Threshold-30)/1800-(Threshold-90)*(Threshold-90)/800)
        <0.001) //30%的情况

        if(abs(((Threshold-30)*(Threshold-30)/1800-(Threshold-90)*(Threshold-90)/800))<0.001)
            printf("Threshold=%f\n", Threshold); //60%的情况
    }
    return 0;
}
```

```
(主/编辑)
// CBestThreshold.cpp : 定义控制台应用程序的入口点。
//
#include "stdafx.h"
#include "math.h"
int _tmain(int argc, _TCHAR* argv[])
{
    double Threshold;
    for(Threshold=0.0;Threshold<255;Threshold=Threshold+0.1)
    {
        if(abs((1.946+(Threshold-30)*(Threshold-30)/1800-(Threshold-90)*(Threshold-90)/800))<0.005)
        //if(abs(((Threshold-30)*(Threshold-30)/1800-(Threshold-90)*(Threshold-90)/800))<0.001)
        printf("Threshold=%f\n",Threshold);
    }
    return 0;
}
```



```
// CBestThreshold.cpp : 定义控制台应用程序的入口点。
//
#include "stdafx.h"
#include "math.h"
int _tmain(int argc, _TCHAR* argv[])
{
    double Threshold;
    for(Threshold=0.0;Threshold<255;Threshold=Threshold+0.1)
    {
        //if(abs((1.946+(Threshold-30)*(Threshold-30)/1800-(Threshold-90)*(Threshold-90)/800))<0.001)
        if(abs(((Threshold-30)*(Threshold-30)/1800-(Threshold-90)*(Threshold-90)/800))<0.001)
        printf("Threshold=%f\n",Threshold);
    }
    return 0;
}
```



排除较高的解，两种情况的解分别为 48 和 66，与前面的计算结果一致，但是感觉自己绕了个圈子，好像把简单的问题复杂了。

参考：

1. <http://wenku.baidu.com/link?url=sTKgmScg2ozr5A5PbTB26uFT6eBT381Lx6zHRloSjjAvlYzEPX64KqLXZpw0Cr0d-wLNpc5k2tHhNWMb7TGWKwtRp3K3gaqDuf43bu4Wy9u>

2. [计算区域灰度值方差](#)